



# Preppy

## Easy, unlimited text-preprocessing

### Features

#### Easy programming

- Unlimited flexibility
- Uses Python, an easy OO language

#### Availability

- Open Source
- Most flexible FreeBSD license
- Available now

#### Platform-independence

- Windows XP, 2000, NT, ME, 98, 95
- Mac OS 8.x, OS 9.x, OS X 10.x
- Virtually any Unix

#### Static vs. dynamic text

Software developers regularly encounter the situation of dealing with large amounts of static text in which some short parts need to be dynamically filled-in with data coming from external sources like databases, CGI requests or just other text files, flat ones or XML.

This is actually so common that many solutions have been invented to tackle this problem. Some of them are JSP, ASP, DTML, PHP and many more. All of them provide some kind of mechanism for embedding executable code into text files that is usually expanded in some moment of time with data fetched from somewhere.

Unfortunately, many of these solutions offer limited capabilities, e.g. because they restrict the complexity of statements you're allowed to embed to functions only, say. This is either for reasons of the used programming language itself or out of ideologic concerns, arguing that you should not make excessiv use of this kind of preprocessing anyway.

#### Preppy offers unlimited generality

Preppy, on the other hand, offers you not only functions or loops and conditions, but any arbitrary programming code to embed in your files. It does so by providing a mechanism to place executable code inside (configurable) tags, by default double curly braces. As Preppy is a single Python module, the code you can embed also needs to be Python. Fortunately, this is so easy that you probably know much of it already, even if you have never written a single line before!

But in addition to simple control flow statements like loops and conditions, you can make use of any Python feature you like in your embedded code. E.g. namespaces, code imported from the Python standard library or any other external modules.

#### License and availability

For several years now ReportLab makes heavy use itself of Preppy in serious large-scale time-critical applications in the financial and other industries. Over the last year Preppy has been used almost unchanged so there is good reason to regard it as a solid tool. Being a small Python module ReportLab has decided to put it under the FreeBSD license and "give it away" for free, in the hope that others might also find it useful.



## Features

## Benefits

---

### Easy programming

---

Unlimited flexibility

- Define your own start/end tags
- Embed code of arbitrary complexity

Uses Python, an easy OO language

- Profit from the easiest to learn and use RAD (Rapid Application Development) language
- Understand code (yours and others') months after writing
- Enjoy a modern, simple and powerful object paradigm

---

### Availability

---

Open Source

- Fully available source code
- Forget hidden APIs and undocumented features
- Adapt the code to your particular needs if needed

Most flexible FreeBSD license

- Do with it whatever you like (just mention the copyright)
- Use it for making as much profit as you like/can
- Develop applications instead of license headaches

Available now

- Download it right now from <ftp://ftp.reportlab.com/preppy.zip>

---

### Platform-independence

---

Windows XP, 2000, NT, ME, 98, 95

- Runs unchanged on any Windows
- Be independent of magic Windows DLLs

Mac OS 8.x, OS 9.x, OS X 10.x

- Runs unchanged on any MacOS where MacPython is available (probably all)
- Postpone your migration decision to OS X if needed

Virtually any Unix

- Runs unchanged on any flavour of Unix where Python compiles (probably all)
- Probably also runs on more exotic platforms like AS/400, OS/2, BeOS, Sony PlayStation and Windows CE, but has not been tested there, yet



## Who should use this?

- Web publishers creating complex documents in HTML or XML that contain large amounts of conditional text.
- Developers who need to enrich static text with dynamic elements using the unlimited power of a full-fledged programming language.

## System Requirements

- Python versions 1.5.2 or above (2.2 being the most recent).

## Support and Related Services

Preppy comes with one electronic document in HTML format.

Online support is provided by ReportLab employees and consultants on an as-time-permits basis in the mailinglist [reportlab-users@yahoogroups.com](mailto:reportlab-users@yahoogroups.com). Hundreds of other end users are also present there and can help with any questions.

For more information, visit [www.reportlab.com](http://www.reportlab.com).

## The Making of...

These datasheets and white papers are, of course, generated from ReportLab's own software. Here's how we do it:

- we defined a little XML document type to hold the required information for product datasheets and white papers.
- we are then free to use many of the excellent XML editing tools such as XML Spy ([www.xmlspy.com](http://www.xmlspy.com)) to input the data. These check that the data complies with the required format as you type.
- we designed a template in a mixture of static RML and short dynamic expressions in our PREPPY preprocessor (much like JSP or ASP). This is compiled on first use (or whenever the design changes) into a program which can merge the dynamic input and create a "fully-expanded" RML document.
- drawings and charts come from our Diagra Charting and Graphics framework. As well as generating crisp vector drawings in PDF, Diagra can output EPS for professional publishing and bitmaps for the web.
- any "page miniatures" you see are brought in using our PageCatcher product, which lets us import existing content from other PDF documents. A single tag in the input tells it which page to load from which PDF. These are the exact pages from the sample application – zoom in and look! (*This does make the PDF files larger*).
- RML2PDF then converts the fully expanded RML into a PDF document.

The program can be run on demand, but is also run on our web server each night to regenerate all the documents. And we could easily produce HTML versions from the original XML if we wished.

## For More Information

For any further technical or developer information, call +1-732-246 8426, email [info@reportlab.com](mailto:info@reportlab.com) or visit [www.reportlab.com](http://www.reportlab.com).

**ReportLab Europe Ltd**  
165 The Broadway, Wimbledon  
London SW19 1NE, UK  
Tel: +44 (0) 20 8544 8049  
Fax: +44 (0) 20 8544 8047  
[www.reportlab.com](http://www.reportlab.com)  
[info@reportlab.com](mailto:info@reportlab.com)

© 2003 ReportLab, Inc. All rights reserved. ReportLab and the ReportLab logo are trademarks of ReportLab, registered in the U.S. and other countries. Other product and company names mentioned herein may be trademarks of their respective companies. Mention of non-ReportLab products is for informational purposes only and constitutes neither an endorsement nor a recommendation. ReportLab assumes no responsibility with regard to the selection, performance, or use of these products. All understandings, agreements, or warranties, if any, place directly between the vendors and the prospective users. Product specifications are subject to change without notice. January 2003.